

[illegible]

EXE

Mod

EDT

ED

ED
EDED
EDED
EDED
ED

ED

ED

ED

ED

ED
EDSYN
LBA

110

```

LL               IIIII
LL               IIIII
LL               II
LL               II
LL               II
LL               II
LL               II
LL               II
LL               II
LL               II
LL               II
LL               II
LL               II
LL               II
LLLLLLLLLLLLL.  IIIII
LLLLLLLLLLLLL  IIIII

SSSSSSSS
SSSSSSSS
SS
SS
SS
SS
SSSSSS
SSSSSS
SS
SS
SS
SS
SSSSSSSS
SSSSSSSS

```



```
0001 0 %TITLE 'EDT$SETCOMMAN - set command file name'
0002 0 MODULE EDT$SETCOMMAN ( ! Set command file name
0003 0 IDENT = 'V04-000' ! File: SETCOMMAN.BLI Edit: JBS1006
0004 0 ) =
0005 1 BEGIN
0006 1
0007 1 *****
0008 1 *
0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0011 1 * ALL RIGHTS RESERVED.
0012 1 *
0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0018 1 * TRANSFERRED.
0019 1 *
0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0022 1 * CORPORATION.
0023 1 *
0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0026 1 *
0027 1 *****
0028 1
0029 1
0030 1
0031 1 ++
0032 1 FACILITY: EDT -- The DEC Standard Editor
0033 1
0034 1 ABSTRACT:
0035 1
0036 1 Set help file name.
0037 1
0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0039 1
0040 1 AUTHOR: John Sauter, CREATION DATE: June 3, 1982
0041 1
0042 1 MODIFIED BY:
0043 1
0044 1 1-001 - Original. JBS 07-Jun-1982
0045 1 1-002 - Don't open the file on the INCLUDE channel if the COMMAND channel
0046 1 is not already open. This is because the initial opens from
0047 1 EDT$EDIT must have a default name which is only supplied on
0048 1 the COMMAND channel. JBS 08-Jun-1982
0049 1 1-003 - Close the correct file. JBS 11-Jun-1982
0050 1 1-004 - Set EDT$SG_CMD_OPN when the command file is open. JBS 11-Jun-1982
0051 1 1-005 - Pass the default file name in the RHB parameter. JBS 15-Jun-1982
0052 1 1-006 - Accept the default name as a parameter. JBS 23-Aug-1982
0053 1 --
0054 1
```

EDT\$SETCOMMAN
V04-000

EDT\$SETCOMMAN - set command file name
Declarations

F 3
16-Sep-1984 01:50:25
14-Sep-1984 12:24:45

VAX-11 Bliss-32 V4.0-742
[EDT.SRC]SETCOMMAN.BLI;1

Page 2
(2)

```
: 56      0055 1 %SBTTL 'Declarations'
: 57      0056 1
: 58      0057 1 : TABLE OF CONTENTS:
: 59      0058 1 :
: 60      0059 1
: 61      0060 1 REQUIRE 'EDTSRC:TRAROUNAM';
: 62      0499 1
: 63      0500 1 FORWARD ROUTINE
: 64      0501 1     EDT$$SET_COMFNAM;
: 65      0502 1
: 66      0503 1 :
: 67      0504 1 : INCLUDE FILES:
: 68      0505 1 :
: 69      0506 1
: 70      0507 1 REQUIRE 'EDTSRC:EDTREQ';
: 71      0642 1
: 72      L 0643 1 %IF %BLISS (BLISS32)
: 73      0644 1 %THEN
: 74      0645 1
: 75      0646 1 REQUIRE 'EDTSRC:SYSSYM';
: 76      0676 1
: 77      0677 1 %FI
: 78      0678 1
: 79      0679 1 :
: 80      0680 1 : MACROS:
: 81      0681 1
: 82      0682 1 :     NONE
: 83      0683 1
: 84      0684 1 : EQUATED SYMBOLS:
: 85      0685 1
: 86      0686 1 :     NONE
: 87      0687 1
: 88      0688 1 : OWN STORAGE:
: 89      0689 1
: 90      0690 1 :     NONE
: 91      0691 1
: 92      0692 1 : EXTERNAL REFERENCES:
: 93      0693 1
: 94      0694 1 :     In the routine
```



```

: 96      0695 1 %SBTTL 'EDT$$SET_COMFNAM - set command file name'
: 97      0696 1
: 98      0697 1 GLOBAL ROUTINE EDT$$SET_COMFNAM (
: 99      0698 1     NADDR,
100      0699 1     NLEN,
101      0700 1     DADDR,
102      0701 1     DLEN
103      0702 1 ) =
104      0703 1
105      0704 1 ++
106      0705 1 FUNCTIONAL DESCRIPTION:
107      0706 1
108      0707 1     Set a new command file name. If the file does not exist there is no
109      0708 1     effect. Otherwise the current command file is abandoned and the new
110      0709 1     command file is read instead.
111      0710 1
112      0711 1 FORMAL PARAMETERS:
113      0712 1
114      0713 1     NADDR      Address of new command file name, or 0 if none.
115      0714 1
116      0715 1     NLEN      Length of new command file name, or 0 if none.
117      0716 1
118      0717 1     DADDR      Address of new command file default name, or 0 if none.
119      0718 1
120      0719 1     DLEN      Length of new command file default name, or 0 if none.
121      0720 1
122      0721 1 IMPLICIT INPUTS:
123      0722 1
124      0723 1     EDT$$A_CMD_NAM
125      0724 1     EDT$$G_CMD_NAMLEN
126      0725 1     EDT$$G_CMD_OPN
127      0726 1
128      0727 1 IMPLICIT OUTPUTS:
129      0728 1
130      0729 1     EDT$$A_CMD_NAM
131      0730 1     EDT$$G_CMD_NAMLEN
132      0731 1     EDT$$G_CMD_OPN
133      0732 1
134      0733 1 ROUTINE VALUE:
135      0734 1
136      0735 1     0 = File not present, EDT$$A_CMD_NAM unchanged (or error, accompanied by message)
137      0736 1     1 = file present, EDT$$A_CMD_NAM changed, or no file named.
138      0737 1
139      0738 1 SIDE EFFECTS:
140      0739 1
141      0740 1     NONE
142      0741 1
143      0742 1 --
144      0743 1
145      0744 2 BEGIN
146      0745 2
147      0746 2 EXTERNAL ROUTINE
148      0747 2     EDT$$DEA_HEAP : NOVALUE,
149      0748 2     EDT$$ALO_HEAP,
150      0749 2     EDT$$FMT_MSG : NOVALUE,
151      0750 2     EDT$$CALCFIO,
152      0751 2     EDT$$FIOPN_ERR : NOVALUE;
```

! Set command file name
! Address of new command file name
! Length of new command file name
! Address of new command file default name
! Length of new command file default name

! Deallocate heap storage
! Allocate heap storage
! Put the text for a message in the format buffer
! Do file I/O
! Print a message if a file operation files

```
153      0752 2
154      L 0753 2 %IF %BLISS (BLISS32)
155      0754 2 %THEN
156      0755 2
157      0756 2     EXTERNAL ROUTINE
158      0757 2     STR$FREE1_DX;                ! Deallocate a synamic string
159      0758 2
160      0759 2 %FI
161      0760 2
162      0761 2     EXTERNAL
163      0762 2     EDT$A_CMD_NAM : REF BLOCK [, BYTE],    ! Name of the command file
164      0763 2     EDT$G_CMD_NAMLEN,                    ! Length of the command file name
165      0764 2     EDT$G_CMD_OPN,                        ! 1 = command file is open
166      0765 2     EDT$G_IOFI_NFND;                     ! 1 = last open failed because of file not found
167      0766 2
168      0767 2     EXTERNAL LITERAL
169      0768 2     EDT$K_INCLUDE_FILE,                  ! Channel for testing for presence of command file
170      0769 2     EDT$K_COMMAND_FILE,                  ! Command file channel
171      0770 2     EDT$K_OPEN_INPUT,                    ! Code for opening for input
172      0771 2     EDT$K_CLOSE;                          ! Code for closing a file
173      0772 2
174      0773 2     LOCAL
175      0774 2     FILE_DESC : BLOCK [8, BYTE],
176      0775 2     RHB_DESC : BLOCK [8, BYTE],
177      0776 2     STATUS;
178      0777 2
179      0778 2     MESSAGES ((INSMEM, COMFILCLO, COMFILNOP, COMFILNEX));
180      0779 2
181      L 0780 2 %IF %BLISS (BLISS32)
182      0781 2 %THEN
183      0782 2     FILE_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
184      0783 2     FILE_DESC [DSC$B_CLASS] = DSC$K_CLASS_D;
185      0784 2     RHB_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
186      0785 2     RHB_DESC [DSC$B_CLASS] = DSC$K_CLASS_D;
187      0786 2 %FI
188      0787 2
189      0788 2     FILE_DESC [DSC$W_LENGTH] = 0;
190      0789 2     FILE_DESC [DSC$A_POINTER] = 0;
191      0790 2     RHB_DESC [DSC$W_LENGTH] = 0;
192      0791 2     RHB_DESC [DSC$A_POINTER] = 0;
193      0792 2     !+
194      0793 2     !- Set up RHB_DESC
195      0794 2     !-
196      0795 2     STRING_DESC (RHB_DESC, DLEN, .DADDR);
197      0796 2     !+
198      0797 2     !- Switch to the new command file, if it exists.
199      0798 2     !-
200      0799 2
201      0800 2     IF ((.NADDR NEQA 0) AND .EDT$G_CMD_OPN)
202      0801 2     THEN
203      0802 2     BEGIN
204      0803 2     STRING_DESC (FILE_DESC, NLEN, .NADDR);
205      0804 2     STATUS = EDT$SCALFIO (EDT$K_OPEN_INPUT, EDT$K_INCLUDE_FILE, FILE_DESC, RHB_DESC);
206      0805 2
207      0806 2     IF ( NOT .STATUS)
208      0807 2     THEN
209      0808 2     BEGIN
```



```
210      0809 4
211      0810 4          IF ( NOT .EDT$G_IOFI_NFND) THEN EDT$FIOPN_ERR (EDT$_COMFILNOP, FILE_DESC);
212      0811 4
213      L 0812 4 %IF %BLISS (BLISS32)
214      0813 4 %THEN
215      0814 4          STR$FREE1_DX (FILE_DESC);
216      0815 4          STR$FREE1_DX (RHB_DESC);
217      0816 4 %FI
218      0817 4
219      0818 5          RETURN (0)
220      0819 3          END;
221      0820 3
222      0821 3 !+
223      0822 3 ! Now that we know that the file exists we can close it on the INCLUDE channel.
224      0823 3 !-
225      0824 3          STATUS = EDT$CALLFIO (EDT$_CLOSE, EDT$_INCLUDE_FILE, 0, 0);
226      0825 3
227      0826 4          IF ( NOT .STATUS)
228      0827 3          THEN
229      0828 4          BEGIN
230      0829 4          EDT$FIOPN_ERR (EDT$_COMFILCLO, FILE_DESC);
231      0830 4
232      L 0831 4 %IF %BLISS (BLISS32)
233      0832 4 %THEN
234      0833 4          STR$FREE1_DX (FILE_DESC);
235      0834 4          STR$FREE1_DX (RHB_DESC);
236      0835 4 %FI
237      0836 4
238      0837 4          RETURN (0);
239      0838 3          END;
240      0839 3
241      0840 2          END;
242      0841 2
243      0842 2 !+
244      0843 2 ! Either the file exists, or no command file is currently open, or no file name is specified.
245      0844 2 ! If the command file is currently open, close it.
246      0845 2 !-
247      0846 2
248      0847 2          IF .EDT$G_CMD_OPN
249      0848 2          THEN
250      0849 3          BEGIN
251      0850 3          STRING_DESC (FILE_DESC, EDT$G_CMD_NAMLEN, .EDT$A_CMD_NAM);
252      0851 3          STATUS = EDT$CALLFIO (EDT$_CLOSE, EDT$_COMMAND_FILE, 0, 0);
253      0852 3
254      0853 4          IF ( NOT .STATUS)
255      0854 3          THEN
256      0855 4          BEGIN
257      0856 4          EDT$FIOPN_ERR (EDT$_COMFILCLO, FILE_DESC);
258      0857 4
259      L 0858 4 %IF %BLISS (BLISS32)
260      0859 4 %THEN
261      0860 4          STR$FREE1_DX (FILE_DESC);
262      0861 4          STR$FREE1_DX (RHB_DESC);
263      0862 4 %FI
264      0863 4
265      0864 4          RETURN (0);
266      0865 3          END;
```

```
267 0866
268 0867      EDT$G_CMD_OPN = 0;
269 0868
270 0869      !+ Deallocate the previous string, if any. Note that if EDT$G_CMD_OPN is zero
271 0870      !- the old space is not deallocated. EDT$EDIT takes advantage of this.
272 0871
273 0872
274 0873      IF (.EDT$G_CMD_NAMLEN NEQ 0) THEN EDT$DEA_HEAP (EDT$G_CMD_NAMLEN, EDT$A_CMD_NAM);
275 0874
276 0875      END;
277 0876
278 0877      !+
279 0878      !- If no new file has been specified, clean up and exit.
280 0879
281 0880
282 0881      IF (.NADDR EQLA 0)
283 0882      THEN
284 0883          BEGIN
285 0884              EDT$G_CMD_NAMLEN = 0;
286 0885              EDT$A_CMD_NAM = 0;
287 0886          END
288 0887      ELSE
289 0888          BEGIN
290 0889      !+
291 0890      !- The command file is not open and a file has been specified. The file may
292 0891      !- not be present. Open errors return an error code but print no message.
293 0892      !- There should be an open error only if this is the initial open.
294 0893
295 0894          STRING_DESC (FILE_DESC, NLEN, .NADDR);
296 0895          STATUS = EDT$SCALFIO (EDT$K_OPEN_INPUT, EDT$K_COMMAND_FILE, FILE_DESC, RHB_DESC);
297 0896
298 0897          IF ( NOT .STATUS)
299 0898          THEN
300 0899              BEGIN
301 0900
302 0901      L 0901      4 %IF %BLISS (BLISS32)
303 0902      4 %THEN
304 0903              STR$FREE1_DX (FILE_DESC);
305 0904              STR$FREE1_DX (RHB_DESC);
306 0905      4 %FI
307 0906
308 0907              RETURN (0);
309 0908          END;
310 0909
311 0910      !+
312 0911      !- Remember the new file name for when we CLOSE it.
313 0912
314 0913
315 0914          IF EDT$ALO_HEAP (NLEN, EDT$A_CMD_NAM)
316 0915          THEN
317 0916              BEGIN
318 0917                  EDT$G_CMD_NAMLEN = .NLEN;
319 0918                  EDT$SCPY_MEM (.NLEN, .NADDR, .EDT$A_CMD_NAM);
320 0919              END
321 0920          ELSE
322 0921              BEGIN
323 0922                  EDT$FMT_MSG (EDT$_INSMEM);
```



```

: 324      0923  4
: 325      L 0924  4 %IF %BLISS (BLISS32)
: 326      0925  4 %THEN
: 327      0926      STR$FREE1_DX (FILE_DESC);
: 328      0927  4      STR$FREE1_DX (RHB_DESC);
: 329      0928  4 %FI
: 330      0929  4
: 331      0930      RETURN (0);
: 332      0931  3      END;
: 333      0932  3
: 334      0933  3      EDT$$G_CMD_OPN = 1;
: 335      0934  2      END;
: 336      0935  2
: 337      L 0936  2 %IF %BLISS (BLISS32)
: 338      0937  2 %THEN
: 339      0938      STR$FREE1_DX (FILE_DESC);
: 340      0939  2      STR$FREE1_DX (RHB_DESC);
: 341      0940  2 %FI
: 342      0941  2
: 343      0942      RETURN (1);
: 344      0943  1      END;
```

! of routine EDT\$\$SET_COMFNAM

.TITLE EDT\$SETCOMMAN EDT\$SETCOMMAN - set command file
name

.IDENT \V04-000\

```
.EXTRN EDT$$DEA_HEAP, EDT$$ALO_HEAP
.EXTRN EDT$$FMT_MSG, EDT$$CALLFIO
.EXTRN EDT$$FIOPN_ERR, STR$FREE1_DX
.EXTRN EDT$$A_CMD_NAM, EDT$$G_CMD_NAMLEN
.EXTRN EDT$$G_CMD_OPN, EDT$$G_IOFI_NFND
.EXTRN EDT$K_INCLUDE_FILE
.EXTRN EDT$K_COMMAND_FILE
.EXTRN EDT$K_OPEN_INPUT
.EXTRN EDT$K_CLOSE, EDT$ INSMEM
.EXTRN EDT$ COMFILCLO, EDT$ COMFILNOP
.EXTRN EDT$ COMFILNEX, STR$COPY_R
```

.PSECT _EDT\$CODE,NOWRT, SHR, PIC,2

```
.ENTRY EDT$$SET_COMFNAM, Save R2,R3,R4,R5,R6,R7,- ; 0697
R8,R9,R10,R11
MOVAB STR$COPY_R, R11
MOVAB STR$FREE1_DX, R10
MOVAB EDT$$CALLFIO, R9
MOVAB EDT$$G_CMD_OPN, R8
MOVAB EDT$$G_CMD_NAMLEN, R7
MOVAB EDT$$A_CMD_NAM, R6
SUBL2 #12, SP
MOVL #34471936, FILE_DESC ; 0788
CLRL FILE_DESC+4 ; 0789
PUSHL #34471936 ; 0790
CLRL RHB_DESC+4 ; 0791
PUSHL DADDR ; 0795
PUSHAB DLEN
PUSHAB RHB_DESC
```

```
OFFC 00000
5B 00000000G 00 9E 00002
5A 00000000G 00 9E 00009
59 00000000G 00 9E 00010
58 00000000G 00 9E 00017
57 00000000G 00 9E 0001E
56 00000000G 00 9E 00025
5E 00000000G 0C C2 0002C
04 AE 020E0000 8F D0 0002F
08 AE D4 00037
020E0000 8F DD 0003A
04 AE D4 00040
0C AC DD 00043
10 AC 9F 00046
08 AE 9F 00049
```

6B		03	FB	0004C	CALLS	#3, STR\$COPY_R	
52	04	AC	DO	0004F	MOVL	NADDR, R2	0800
		51	13	00053	BEQL	2\$	
4E		68	E9	00055	BLBC	EDT\$\$G_CMD_OPN, 2\$	
		52	DD	00058	PUSHL	R2	0803
	08	AC	9F	0005A	PUSHAB	NLEN	
	10	AE	9F	0005D	PUSHAB	FILE_DESC	
6B		03	FB	00060	CALLS	#3, STR\$COPY_R	
		5E	DD	00063	PUSHL	SP	0804
	0C	AE	9F	00065	PUSHAB	FILE_DESC	
	00000000G	8F	DD	00068	PUSHL	#EDT\$K_INCLUDE_FILE	
	00000000G	8F	DD	0006E	PUSHL	#EDT\$K_OPEN_INPUT	
69		04	FB	00074	CALLS	#4, EDT\$\$CALLFIO	
53		50	DO	00077	MOVL	R0, STATUS	
12		53	E8	0007A	BLBS	STATUS, 1\$	0806
56	00000000G	00	E8	0007D	BLBS	EDT\$\$G_IOFI_NFND, 5\$	0810
	08	AE	9F	00084	PUSHAB	FILE_DESC	
	00000000G	8F	DD	00087	PUSHL	#EDT\$COMFILNOP	
		44	11	0008D	BRB	4\$	
		7E	7C	0008F	CLRQ	-(SP)	0824
	00000000G	8F	DD	00091	PUSHL	#EDT\$K_INCLUDE_FILE	
	00000000G	8F	DD	00097	PUSHL	#EDT\$K_CLOSE	
69		04	FB	0009D	CALLS	#4, EDT\$\$CALLFIO	
53		50	DO	000A0	MOVL	R0, STATUS	
24		53	E9	000A3	BLBC	STATUS, 3\$	0826
44		68	E9	000A6	BLBC	EDT\$\$G_CMD_OPN, 7\$	0847
		66	DD	000A9	PUSHL	EDT\$\$A_CMD_NAM	0850
		57	DD	000AB	PUSHL	R7	
	10	AE	9F	000AD	PUSHAB	FILE_DESC	
6B		03	FB	000B0	CALLS	#3, STR\$COPY_R	
		7E	7C	000B3	CLRQ	-(SP)	0851
	00000000G	8F	DD	000B5	PUSHL	#EDT\$K_COMMAND_FILE	
	00000000G	8F	DD	000BB	PUSHL	#EDT\$K_CLOSE	
69		04	FB	000C1	CALLS	#4, EDT\$\$CALLFIO	
53		50	DO	000C4	MOVL	R0, STATUS	
12		53	E8	000C7	BLBS	STATUS, 6\$	0853
	08	AE	9F	000CA	PUSHAB	FILE_DESC	0856
	00000000G	8F	DD	000CD	PUSHL	#EDT\$COMFILCLO	
00000000G	00	02	FB	000D3	CALLS	#2, EDT\$\$FIOPN_ERR	
		6A	11	000DA	BRB	10\$	0860
		68	D4	000DC	CLRL	EDT\$\$G_CMD_OPN	0867
		67	D5	000DE	TSTL	EDT\$\$G_CMD_NAMLEN	0873
		0B	13	000E0	BEQL	7\$	
		56	DD	000E2	PUSHL	R6	
		57	DD	000E4	PUSHL	R7	
00000000G	00	02	FB	000E6	CALLS	#2, EDT\$\$DEA_HEAP	
		52	D5	000ED	TSTL	R2	0881
		06	12	000EF	BNEQ	8\$	
		67	D4	000F1	CLRL	EDT\$\$G_CMD_NAMLEN	0884
		66	D4	000F3	CLRL	EDT\$\$A_CMD_NAM	0885
		5F	11	000F5	BRB	12\$	0881
		52	DD	000F7	PUSHL	R2	0894
	08	AC	9F	000F9	PUSHAB	NLEN	
	10	AE	9F	000FC	PUSHAB	FILE_DESC	
6B		03	FB	000FF	CALLS	#3, STR\$COPY_R	
		5E	DD	00102	PUSHL	SP	0895
	0C	AE	9F	00104	PUSHAB	FILE_DESC	

EDT\$SETCOMMAN
V04-000

EDT\$SETCOMMAN - set command file name
EDT\$\$\$SET_COMFNAM - set command file name

M 3
16-Sep-1984 01:50:25
14-Sep-1984 12:24:45

VAX-11 Bliss-32 V4.0-742
[EDT.SRC]SETCOMMAN.BLI;1

Page 9
(3)

		00000000G	8F	DD	00107	PUSHL	#EDT\$K_COMMAND_FILE	:	
		00000000G	8F	DD	0010D	PUSHL	#EDT\$K_OPEN_INPUT	:	
	69		04	FB	00113	CALLS	#4, EDT\$\$\$CALLFIO	:	
	53		50	D0	00116	MOVL	R0, STATUS	:	
	2A		53	E9	00119	BLBC	STATUS, 10\$	:	0897
			56	DD	0011C	PUSHL	R6	:	0914
		08	AC	9F	0011E	PUSHAB	NLEN	:	
	00000000G	00	02	FB	00121	CALLS	#2, EDT\$\$\$ALO_HEAP	:	
	0E		50	E9	00128	BLBC	R0, 9\$	:	
	67	08	AC	D0	0012B	MOVL	NLEN, EDT\$\$G_CMD_NAMLEN	:	0917
	50		66	D0	0012F	MOVL	EDT\$\$A_CMD_NAM, R0	:	0918
60	62	08	AC	28	00132	MOVC3	NLEN, (R2), (R0)	:	
			1A	11	00137	BRB	11\$	:	0914
		00000000G	8F	DD	00139	PUSHL	#EDT\$ INSMEM	:	0922
	00000000G	00	01	FB	0013F	CALLS	#1, EDT\$\$FMT_MSG	:	
		08	AE	9F	00146	PUSHAB	FILE_DESC	:	0926
	6A		01	FB	00149	CALLS	#1, STR\$FREE1_DX	:	
			5E	DD	0014C	PUSHL	SP	:	0927
	6A		01	FB	0014E	CALLS	#1, STR\$FREE1_DX	:	
			12	11	00151	BRB	13\$	:	0930
	68		01	D0	00153	MOVL	#1, EDT\$\$G_CMD_OPN	:	0933
		08	AE	9F	00156	PUSHAB	FILE_DESC	:	0938
	6A		01	FB	00159	CALLS	#1, STR\$FREE1_DX	:	
			5E	DD	0015C	PUSHL	SP	:	0939
	6A		01	FB	0015E	CALLS	#1, STR\$FREE1_DX	:	
	50		01	D0	00161	MOVL	#1, R0	:	0942
			04	00164	RET			:	
			50	D4	00165	CLRL	R0	:	0943
			04	00167	RET			:	

; Routine Size: 360 bytes, Routine Base: _EDT\$CODE + 0000

; 345 0944 1
; 346 0945 1 !<BLF/PAGE>

EDT\$SETCOMMAN
V04-000

EDT\$SETCOMMAN - set command file name
EDT\$\$\$SET_COMFNAM - set command file name

N 3
16-Sep-1984 01:50:25
14-Sep-1984 12:24:45

VAX-11 Bliss-32 V4.0-742
[EDT.SRC]SETCOMMAN.BLI;1

Page 10
(4)

: 348 0946 1 END
: 349 0947 1
: 350 0948 0 ELUDOM

! of module EDT\$SETCOMMAN

PSECT SUMMARY

Name	Bytes	Attributes
_EDT\$CODE	360	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[EDT.SRC]EDT.L32;1	377	2	0	40	00:00.2
_\$255\$DUA28:[EDT.SRC]PSECTS.L32;1	2	1	50	7	00:00.1
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	6	0	581	00:04.0

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS\$:SETCOMMAN/OBJ=OBJ\$:SETCOMMAN MSRC\$:SETCOMMAN.BLI/UPDATE=(ENH\$:SETCOMMAN)

: Size: 360 code + 0 data bytes
: Run Time: 00:25.5
: Elapsed Time: 00:37.3
: Lines/CPU Min: 2227
: Lexemes/CPU-Min: 7346
: Memory Used: 152 pages
: Compilation Complete

0140

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY